UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/731,805	12/08/2003	Christoph Bussler	021756-002600US	5730

51206 7590 02/19/2010

TOWNSEND AND TOWNSEND AND CREW LLP/ORACLE
TWO EMBARCADERO CENTER
8TH FLOOR
SAN FRANCISCO, CA 94111-3834

EXAMINER	
SYED, FARHAN M	

ART UNIT	PAPER NUMBER
2165	

MAIL DATE	DELIVERY MODE
02/19/2010	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Office Action Summary	Application No.	Applicant(s)	
	10/731,805	BUSSLER ET AL.	
	Examiner	Art Unit	
	FARHAN M. SYED	2165	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on 22 October 2009.

2a)☒ This action is **FINAL**. 2b)☐ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) 1-27 is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) 1-27 is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☐ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____ .
5) ☐ Notice of Informal Patent Application
6) ☐ Other: _____.

DETAILED ACTION

1. Claims 1-27, filed 22 October 2009, are pending. The Examiner acknowledges amended claim 19 and newly added claim 27.

Response to Remarks/Argument

2. Applicant's arguments filed 22 October 2009 have been fully considered but they are not persuasive.

The applicant argues:

(1) The cited art does not teach “applying the validation rules to each object in the list of objects.”

The Examiner disagrees. The combination of Drake, Pastor, and Sang-Kyun teaches applying the validation rules to each object in the list of objects (The Examiner notes that in the data model, validation objects (i.e. rules) are applies to data (i.e. each object in the list of object.)(See Figure 2, see also paragraphs [0011, 0022, and 0024-0028]).

(2) The cited art does not teach “querying the database to retrieve one or more validation rules for each object in the list of objects.”

The Examiner disagrees. The combination of Drake, Pastor, and Sang-Kyun teaches querying the database to retrieve one or more validation rules for each object in the list of objects (Drake, See Fig. 3 and paragraphs [0011, 0019-0020, and 0024-0025] where determining rules for the subject in the context is taught; in addition, the data

model taught in Drake implicitly refers to an abstract database model that is capable of storing and retrieving data. However, Pastor further teaches using SQL (structured query language) to retrieve validated formal specification from a database server., see from column 20, line 58 to column 25, line 67 and column 35, lines 5-65).

(3) The amendment made to claim 19 has been addressed and incorporated into the rejection below.

(4) There is no motivation to combined the cited arts of record that address the limitations of claims 2, 4, 18, and 20-21.

The Examiner disagrees. In response to applicant's argument that there is no suggestion to combine the references, the examiner recognizes that obviousness can only be established by combining or modifying the teachings of the prior art to produce the claimed invention where there is some teaching, suggestion, or motivation to do so found either in the references themselves or in the knowledge generally available to one of ordinary skill in the art. See *In re Fine*, 837 F.2d 1071, 5 USPQ2d 1596 (Fed. Cir. 1988)and *In re Jones*, 958 F.2d 347, 21 USPQ2d 1941 (Fed. Cir. 1992). In this case, Drake is directed to systems and methods for self-contained validation of data model object content. Pastor is directed to an automated software system that captures and converts data objects into formal specifications and validates for correctness and completeness. Sang-Kyun is directed towards immediate and partial validation of data objects in XML databases. All three are analogous art. Mikhailov is directed to form

handling, which complements and enhances the validation techniques as taught in the Drake, Pastor, and Sang-Hyun.

(5) There is no motivation to combine the cited arts of record that address the limitations of claims 7-13, 16, and 2-24.

The Examiner disagrees. In response to applicant's argument that there is no suggestion to combine the references, the examiner recognizes that obviousness can only be established by combining or modifying the teachings of the prior art to produce the claimed invention where there is some teaching, suggestion, or motivation to do so found either in the references themselves or in the knowledge generally available to one of ordinary skill in the art. See *In re Fine*, 837 F.2d 1071, 5 USPQ2d 1596 (Fed. Cir. 1988)and *In re Jones*, 958 F.2d 347, 21 USPQ2d 1941 (Fed. Cir. 1992). In this case, Drake is directed to systems and methods for self-contained validation of data model object content. Pastor is directed to an automated software system that captures and converts data objects into formal specifications and validates for correctness and completeness. Sang-Kyun is directed towards immediate and partial validation of data objects in XML databases. All three are analogous art. Lindenberg is directed to transferring information between a user interface and a database, which complements and enhances the teachings of Drake, Pastor, and Sang-Kyun.

(6) The Examiner has addressed the newly added claim 27 in the rejection below.

.

Claim Rejections - 35 USC § 103

3. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

4. Claims 1, 3, 5, 6, 14, 15, 17, 19, and 25-27 are rejected under 35 U.S.C. 103(a) as being unpatentable over Drake et al (U.S. 2003/0070142 A1 and known hereinafter as Drake), in view of Pastor et al (U.S 6,681,383 B1 and known hereinafter as Pastor)(newly presented), and in further view of Sang-Kyun et al. ("Immediate and Partial Validation Mechanism for the Conflict Resolution of Update Operations in XML Databases." 2002 and known hereinafter as Sang) (previously presented).

Per claim 1, *Drake* discloses a computer-implemented method of validating data in an object model (See Title teaches validation of data model object content. Also see paragraph [0024].), comprising:

identifying a first subject of validation wherein the first subject is one of an object, an attribute, an association and a collection of objects (Fig. 3 illustrates identifying a first subject of validation as social security number where the name attribute identifies "social_security_number". Further see paragraphs [0024-0025 and 0029]);

determining a context of data validation based on the first subject, the context including one of a) the first subject, and b) the first subject and one or more additional subjects (See Fig. 3 and see paragraphs [0024-0025] wherein designing rules specifically for social security numbers is determining a context of data validation based on the first subject.);

querying the database to retrieve one or more validation rules for each object in the list of objects (See Fig. 3 and paragraphs [0011, 0019-0020, and 0024-0025] where determining rules for the subject in the context is taught; in addition, the data model taught in Drake implicitly refers to an abstract database model that is capable of storing and retrieving data.).

Drake does not explicitly disclose applying the determined validation rules to each subject in the context; and the data is metadata in an object model stored in a database. However, *Pastor* discloses applying the determined validation rules to each subject in the context (state transition diagram describes applying the determined validation rules to each subject, where a valid life refers to an appropriate sequence of states that characterizes the correct behavior of the object. That is, the correct behavior proves positive validation.)(see at least column 12, ,lines 20-50); and metadata models and that the metadata is stored in data sources such as databases (conceptual model is in compliant with UML and therefore is obvious that HTML, an illustration of metadata model is used and stored in the database schema.)(See col. 12 lines 53-55, and col. 2 lines 26-27; see Figure 2).

At the time of the invention, it would have been obvious to a person of ordinary skill in the art of data object models to allow users of the system of *Pastor* including applying the determined validation rules to each subject in the context and a database storing metadata to run the validation application of Drake (See col. 12 lines 53-55, and col. 2 lines 26-27). The motivation would have been to provide multiple users access to the validation application of Drake (See *Pastor* col. 3 lines 37-41).

Drake, in view of Pastor, does not explicitly disclose wherein applying the determined validation rules results in one of partially and completely validating the metadata for the object model and wherein applying the determined validation rules occurs prior to deployment of the object model, a deployment of the object model allowing the object model to be used to store data according to the object model and querying the database to retrieve a list of objects requiring validation, the list of objects determined by the first subject of validation and the determined context.

However, Sang discloses partial or full, i.e. complete, validation (See Abstract of page 387 -1st page-, also page 393-394 where partial or complete validation occurs.) and wherein applying the determined validation rules (i.e. validation mechanism)(page 389) occurs prior to deployment of the object model (i.e. "Before we store or update XML documents in applications, we must verify that their structural information is valid…" "Most of XML database systems use deferred and full validation methods when XML documents are updated.)(page 389), a deployment of the object model allowing the object model to be used to store data according to the object model (i.e. "We construct and store a DFA per each element declaration in DTD files." "The extracted

DTD information is stored in the database for validating update operations…"')(page 389); and querying the database to retrieve a list of objects requiring validation (i.e. XML parser checks the whole document for validation, using XQuery that conforms to DTD. Therefore, querying the database (i.e. XML Databases) to retrieve a list of objects (list of objects reside in the whole XML Document) that requires validation.)(page 388-390), the list of objects determined by the first subject of validation and the determined context (The validation technique used in Sang anticipates the list of objects determined by the first subject of validation and context, because Sang's XML validation determines partial or full validation of objects within an XML document)(pages 389-390).

.Both Drake, in view of Pastor, and Sang are directed to object validation in the art of database technology. Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to have combined the cited references because Sang would have provide for performance improvement as seen in Abstract and pages 393-394 – see "4 Performance Evaluation" through "4.3 Analysis of Performance".

Per claim 3, *Drake* discloses the method of claim 1, wherein identifying includes receiving an indication from a user interface module, said indication identifying the first subject (Fig. 1 illustrates graphical user interface interaction. See paragraphs [0031]).

Per claim 5, *Drake* discloses the method of claim 1, wherein identifying includes receiving an update indication identifying the first subject in response to a modification

of the first subject (See paragraph [0032] teaches update indication.).

Per claim 6, *Drake* discloses the method of claim 1, wherein each of the one or more validation rules is one of a correctness type rule and a completeness type rule, a correctness validation rule operable to be applied while partially validating the object model and a completeness validation rule operable to be applied while completely validating the object model (See Fig. 3 and paragraphs [0004 and 0025] where validation is for correctness and/or completeness validation.).

Per claim 14, *Drake* discloses the method of claim 1, wherein determining one or more validation rules includes identifying rules in rule files based on the subject type of each subject to be validated (See rule file in Fig. 3).

Per claim 15, *Drake* discloses the method of claim 14, wherein each rule file is a Java file (See paragraph [0035] where software packages is JavaBeans implying Java files).

Per claim 17, *Drake* discloses a data validation system for validating an object model, comprising:

means for identifying a first subject of validation, wherein the first subject type is one of an object, an attribute, an association and a collection of objects (Fig. 3 illustrates identifying a first subject of validation as social security number where the

name attribute identifies “social_security_number”. Further see paragraphs [0024-0025 and 0029]);

means for determining a context of data validation based on the first subject, the context including one of a) the first subject, and b) the first subject and one or more additional subjects (See Fig. 3 and see paragraphs [0024-0025] wherein designing rules specifically for social security numbers is determining a context of data validation based on the first subject.);

means for querying the database to retrieve one or more validation rules for each object in the list of objects (See Fig. 3 and paragraphs [0011, 0019-0020, and 0024-0025] where determining rules for the subject in the context is taught; in addition, the data model taught in Drake implicitly refers to an abstract database model that is capable of storing and retrieving data.); and

means for applying the determined validation rules to each subject in the context (See Figs. 3 and 4, also paragraphs [0011, 0024-0025, and 0031] teaches applying validation rules.).

Drake does not explicitly disclose applying the determined validation rules to each subject in the context; and the data is metadata in an object model stored in a database. However, *Pastor* discloses applying the determined validation rules to each subject in the context (state transition diagram describes applying the determined validation rules to each subject, where a valid life refers to an appropriate sequence of states that characterizes the correct behavior of the object. That is, the correct behavior proves positive validation.)(see at least column 12, ,lines 20-50); and metadata models

and that the metadata is stored in data sources such as databases (conceptual model is in compliant with UML and therefore is obvious that HTML, an illustration of metadata model is used and stored in the database schema.)(See col. 12 lines 53-55, and col. 2 lines 26-27; see Figure 2).

At the time of the invention, it would have been obvious to a person of ordinary skill in the art of data object models to allow users of the system of *Pastor* including a database storing metadata to run the validation application of Drake (See col. 12 lines 53-55, and col. 2 lines 26-27). The motivation would have been to provide multiple users access to the validation application of Drake (See *Pastor* col. 3 lines 37-41).

Drake, in view of Pastor, does not explicitly disclose wherein the means for applying the determined validation rules provides for both partially and completely validating the metadata for the object model; and means for querying the database to retrieve a list of objects requiring validation, the list of objects determined by the first subject of validation and the determined context. However, Sang discloses partial or full, i.e. complete, validation (See Abstract of page 387 -1st page-, also page 393-394 where partial or complete validation occurs.) and wherein applying the determined validation rules (i.e. validation mechanism)(page 389) occurs prior to deployment of the object model (i.e. “Before we store or update XML documents in applications, we must verify that their structural information is valid…” “Most of XML database systems use deferred and full validation methods when XML documents are updated.)(page 389), a deployment of the object model allowing the object model to be used to store data

according to the object model (i.e. “We construct and store a DFA per each element declaration in DTD files.” “The extracted DTD information is stored in the database for validating update operations…”’)(page 389); and means for querying the database to retrieve a list of objects requiring validation (i.e. XML parser checks the whole document for validation, using XQuery that conforms to DTD. Therefore, querying the database (i.e. XML Databases) to retrieve a list of objects (list of objects reside in the whole XML Document) that requires validation.)(page 388-390), the list of objects determined by the first subject of validation and the determined context (The validation technique used in Sang anticipates the list of objects determined by the first subject of validation and context, because Sang’s XML validation determines partial or full validation of objects within an XML document)(pages 389-390).

Both Drake, in view of Pastor, and Sang are directed to object validation in the art of database technology. Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to have combined the cited references because Sang would have provide for performance improvement as seen in Abstract and pages 393-394 – see "4 Performance Evaluation" through "4.3 Analysis of Performance".

Per claim 19, *Drake* discloses a method of validating data in an object model, the method comprising:

receiving user defined rules, each rule defining a validation rule on a data object (See Fig. 4 illustrates and paragraph 0031 teaches custom validation is user validation. See paragraphs [0011 and 0020] teaches receiving user defined rules. See Title and

paragraph [0024 teaches validation rule on data model object content.), each rule being one of a completeness type rule and a correctness type rule (Fig. 3 illustrates and paragraphs [0005 and 0025] disclose completeness and correctness type rules.);

storing the validation rules (See Fig. 1 illustrates stored validation rules in data model.);

identifying a first subject of metadata validation, wherein the first subject has a subject type selected from the group consisting of is one of an attribute, an association, an object and a collection of objects (Fig. 3 illustrates identifying a first subject of validation as social security number where the name attribute identifies "social_security_number". Further see paragraphs [0024-0025 and 0028-0029].);

determining a context of validation based on the first subject, wherein the context includes the first subject and none, one or more additional subjects (See Fig. 3 and see paragraphs [0024-0025] wherein designing rules specifically for social security numbers is determining a context of data validation based on the first subject.); and

applying a correctness type validation rule to each of the first and additional subjects (See Figs. 3 and 4, also paragraphs [0011, 0024-0025, and 0031] teaches applying validation rules. See Fig. 3 and paragraphs [0004] where validation is for correctness.);

applying a correctness type and completeness type validation rule to each of the determined first and additional subjects (See Fig. 3 and paragraphs [0004 and 0025] where Drake teaches validation for correctness and completeness.)

Drake does not explicitly disclose applying the determined validation rules to each subject in the context; and the data is metadata in an object model stored in a database.

However, *Pastor* discloses applying the determined validation rules to each subject in the context (state transition diagram describes applying the determined validation rules to each subject, where a valid life refers to an appropriate sequence of states that characterizes the correct behavior of the object. That is, the correct behavior proves positive validation.)(see at least column 12, ,lines 20-50);

receiving a selection from the user of a type of validation to perform (i.e. Figure 5 depicts a display that allows receiving a selection from the user of a type of validation to perform.)(Figure 5), the type selected from the group consisting of correctness and completeness (i.e. validation techniques are implemented for correctness and completeness.)(see column 21, line 15 to column 24, line 22); and

metadata models and that the metadata is stored in data sources such as databases (conceptual model is in compliant with UML and therefore is obvious that HTML, an illustration of metadata model is used and stored in the database schema.)(See col. 12 lines 53-55, and col. 2 lines 26-27; see Figure 2).

Drake does not explicitly disclose storing the validation rules to the database. However, Drake teaches storing validation rules in the data model (See Fig. 1-2 and paragraph [0019]), while *Pastor* discloses storing the data model in a database (See col. 12 lines 53-55, and col. 2 lines 26-27).

At the time of the invention, it would have been obvious to a person of ordinary skill in the art of data object models to allow users of the system of *Pastor* including a database storing metadata to run the validation application of Drake (See col. 12 lines 53-55, and col. 2 lines 26-27). The motivation would have been to provide multiple users access to the validation application of Drake (See *Pastor* col. 3 lines 37-41).

Drake, in view of Pastor, does not explicitly disclose wherein applying the determined validation rules for correctness when only a portion of the metadata is determined; and apply for correctness and completeness when all of the metadata is determined. However, Sang discloses partial or full, i.e. complete, validation (See Abstract of page 387 -1st page-, also page 393-394 where partial or complete validation occurs.) and wherein applying the determined validation rules (i.e. validation mechanism)(page 389) occurs prior to deployment of the object model (i.e. “Before we store or update XML documents in applications, we must verify that their structural information is valid…” “Most of XML database systems use deferred and full validation methods when XML documents are updated.)(page 389), a deployment of the object model allowing the object model to be used to store data according to the object model (i.e. “We construct and store a DFA per each element declaration in DTD files.” “The extracted DTD information is stored in the database for validating update operations…”’)(page 389).

Both Drake, in view of Pastor, and Sang are directed to object validation in the art of database technology. Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to have combined the cited references

because Sang would have provide for performance improvement as seen in Abstract and pages 393-394 – see "4 Performance Evaluation" through "4.3 Analysis of Performance".

Per claim(s) 25, rejection of claims 14 and 19 are fully incorporated.

Claim 25 is rejected under the same rationale as claim 14 due to the similarity in scope in the limitations of the claims. (See respective claims above.).

Per claim(s) 26, rejection of claims 15, 19 are fully incorporated. Claim26 is rejected under the same rationale as claim 15 and 19 due to the similarity in scope in the limitations of the claims. (See respective claims above.).

Per claim 27, Drake teaches a computer readable medium containing a set of instructions that when executed by a processor cause the processor to validate metadata in an object model in a database, the set of instructions causing the processor to perform the steps of:

receiving user defined rules, each rule defining a validation rule on a data object (See Fig. 4 illustrates and paragraph 0031 teaches custom validation is user validation. See paragraphs [0011 and 0020] teaches receiving user defined rules. See Title and paragraph [0024 teaches validation rule on data model object content.), each rule being one of a completeness type rule and a correctness type rule (Fig. 3 illustrates and paragraphs [0005 and 0025] disclose completeness and correctness type rules.);

storing the validation rules (See Fig. 1 illustrates stored validation rules in data model.);

identifying a first subject of metadata validation, wherein the first subject has a subject type selected from the group consisting of is one of an attribute, an association, an object and a collection of objects (Fig. 3 illustrates identifying a first subject of validation as social security number where the name attribute identifies “social_security_number”. Further see paragraphs [0024-0025 and 0028-0029].);

determining a context of validation based on the first subject, wherein the context includes the first subject and none, one or more additional subjects (See Fig. 3 and see paragraphs [0024-0025] wherein designing rules specifically for social security numbers is determining a context of data validation based on the first subject.); and

applying a correctness type validation rule to each of the first and additional subjects (See Figs. 3 and 4, also paragraphs [0011, 0024-0025, and 0031] teaches applying validation rules. See Fig. 3 and paragraphs [0004] where validation is for correctness.);

applying a correctness type and completeness type validation rule to each of the determined first and additional subjects (See Fig. 3 and paragraphs [0004 and 0025] where Drake teaches validation for correctness and completeness.)

Drake does not explicitly disclose applying the determined validation rules to each subject in the context; and the data is metadata in an object model stored in a database.

However, *Pastor* discloses applying the determined validation rules to each subject in the context (state transition diagram describes applying the determined validation rules to each subject, where a valid life refers to an appropriate sequence of states that characterizes the correct behavior of the object. That is, the correct behavior proves positive validation.)(see at least column 12, ,lines 20-50);

receiving a selection from the user of a type of validation to perform (i.e. Figure 5 depicts a display that allows receiving a selection from the user of a type of validation to perform.)(Figure 5), the type selected from the group consisting of correctness and completeness (i.e. validation techniques are implemented for correctness and completeness.)(see column 21, line 15 to column 24, line 22); and

metadata models and that the metadata is stored in data sources such as databases (conceptual model is in compliant with UML and therefore is obvious that HTML, an illustration of metadata model is used and stored in the database schema.)(See col. 12 lines 53-55, and col. 2 lines 26-27; see Figure 2).

Drake does not explicitly disclose storing the validation rules to the database. However, Drake teaches storing validation rules in the data model (See Fig. 1-2 and paragraph [0019]), while *Pastor* discloses storing the data model in a database (See col. 12 lines 53-55, and col. 2 lines 26-27).

At the time of the invention, it would have been obvious to a person of ordinary skill in the art of data object models to allow users of the system of *Pastor* including a database storing metadata to run the validation application of Drake (See col. 12 lines

53-55, and col. 2 lines 26-27). The motivation would have been to provide multiple users access to the validation application of Drake (See *Pastor* col. 3 lines 37-41).

Drake, in view of Pastor, does not explicitly disclose wherein applying the determined validation rules for correctness when only a portion of the metadata is determined; and apply for correctness and completeness when all of the metadata is determined. However, Sang discloses partial or full, i.e. complete, validation (See Abstract of page 387 -1st page-, also page 393-394 where partial or complete validation occurs.) and wherein applying the determined validation rules (i.e. validation mechanism)(page 389) occurs prior to deployment of the object model (i.e. “Before we store or update XML documents in applications, we must verify that their structural information is valid…” “Most of XML database systems use deferred and full validation methods when XML documents are updated.)(page 389), a deployment of the object model allowing the object model to be used to store data according to the object model (i.e. “We construct and store a DFA per each element declaration in DTD files.” “The extracted DTD information is stored in the database for validating update operations…”')(page 389).

Both Drake, in view of Pastor, and Sang are directed to object validation in the art of database technology. Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to have combined the cited references because Sang would have provide for performance improvement as seen in Abstract and pages 393-394 – see "4 Performance Evaluation" through "4.3 Analysis of Performance".

5. Claims 2, 4, 18, and 20-21 are rejected under 35 U.S.C. 103(a) as being unpatentable over Drake et al. (US Pub. No. 2003/0070142 A1) (previously presented), hereinafter *Drake*, in view of Pastor et al (U.S 6,681,383 B1 and known hereinafter as Pastor)(newly presented), in view of Sang-Kyun et al. (“Immediate and Partial Validation Mechanism for the Conflict Resolution of Update Operations in XML Databases.” 2002) (previously presented), hereinafter Sang, and in further view of Mikhailov et al. (USPN 6,968,500 B2) (previously presented), hereinafter *Mikhailov*.

Per claim 2, the rejection of claim 1 is incorporated. Furthermore, *Drake* as modified by *Rasmussen* does not explicitly disclose the method of claim 1, wherein each subject is a metadata object selected from the group consisting of a MetaAttribute, a MetaAssociation, a MetaAssociationEnd, a MetaClass and a MetaCollection. However, *Mikhailov* discloses a group of types of metadata associated with the corresponding database table (See col. 5 lines 39-54, col. 14 lines 62-64, and col. 1 lines 1-31.)

At the time of the invention, it would have been obvious to a person of ordinary skill in the art of data object models to allow users of the application and system of *Drake* and *Rasmussen* to utilize the group of types of metadata associated by *Mikhailov* (See col. 5 lines 39-54, col. 14 lines 62-64, and col. 1 lines 1-31.). The motivation would have been to provide multiple users access to the validation application of Drake (See *Pastor* col. 3 lines 37-41) with the convenience of structured online communication as taught by *Mikhailov* (See col. 1 lines 27-31).

Per claim 4, the rejection of claim 1 is incorporated. Furthermore, *Drake* and *Pastor* do not explicitly disclose the method of claim 1, wherein identifying includes receiving an indication from a configuration management module, said indication identifying the first subject. However, *Mikhailov* discloses automation forms handling application service (See col. 5 lines 19-38.).

At the time of the invention, it would have been obvious to a person of ordinary skill in the art of data object models to allow users of the application and system of *Drake* and *Pastor* to utilize the group of types of metadata associated by *Mikhailov* (See col. 5 lines 39-54, col. 14 lines 62-64, and col. 1 lines 1-31.). The motivation would have been to provide multiple users access to the validation application of Drake (See *Pastor* col. 3 lines 37-41) with the convenience of structured online communication as taught by *Mikhailov* (See col. 1 lines 27-31).

Per claim(s) 18 and 20, rejection of claims 2, 17, and 19 are fully incorporated. Claims 18 and 20 are rejected under the same rationale as claim 2 due to the similarity in scope in the limitations of the claims. (See respective claims above.).

Per claim(s) 21, rejection of claims 3, 4, and 19 are fully incorporated. Claim 21 is rejected under the same rationale as claims 3 and 4 due to the similarity in scope in the limitations of the claims. (See respective claims above.).

6. Claims 7-13, 16, and 22-24 are rejected under 35 U.S.C. 103(a) as being unpatentable over Drake et al. (US Pub. No. 2003/0070142 A1) (previously presented), hereinafter *Drake*, in view of Pastor et al (U.S 6,681,383 B1 and known hereinafter as Pastor)(newly presented), hereinafter *Rasmussen*, in view of Sang-Kyun et al. ("Immediate and Partial Validation Mechanism for the Conflict Resolution of Update Operations in XML Databases." 2002) (previously presented), hereinafter Sang, and in further view of Lindberg et al. (US Pub. No. 2003/0028540 A1) (previously presented), hereinafter *Lindberg*.

Per claim 7, the rejection of claim 1 is incorporated. Furthermore, *Drake* as modified by *Pastor* does not explicitly disclose the method of claim 1, wherein the first subject is a root object for a collection of associated objects. However, *Lindberg* discloses a first subject as a root object for a collection of associated objects (See Fig. 2 shows "Person" as the root. Also see paragraph [0016]).

At the time of the invention, it would have been obvious to a person of ordinary skill in the art of data object models to allow users of the application and system of *Drake* and *Pastor* where social security number is the subject to utilize the method of associating objects as taught by *Lindberg* (See Fig. 2.), whereby the person could be associated with other objects by its social security number attribute. The motivation would have been to provide a more useful way of organizing and labeling the data without changing the information model layer (*Lindberg* paragraph [0016]) and thereby

allowing for easily and efficiently sharing the data and data validation among multiple presentations (*Drake* paragraph [0022]).

Per claims 8, the rejection of claim 7 is incorporated. *Drake, Pastor,* and *Lindberg* discloses wherein the collection of objects is a deployable collection including all objects transitively associated with the root object (See paragraph [0021] of *Drake.)* and (See Fig. 2 of *Lindberg*).

Per claim 9, the rejection of claim 7 is incorporated. *Drake in view of Pastor* and *Lindberg* discloses wherein the collection of objects is an aggregated collection including the root object and all of its strongly aggregated child objects recursively (See Fig. 2 of *Lindberg*).

Per claim 10, the rejection of claims 1 and 7 are incorporated, wherein determining a context includes:

a) traversing all associations with a root object to identify target objects (See Fig. 2 of *Lindberg*);

b) repeating a) for each target object, with each target object as the root object (See Fig. 2 of *Lindberg*); and

c) generating a list of all target objects, wherein said list of objects represents a transitive closure based on the root object (See paragraph [0021] of *Drake* and See Fig. 2 of *Lindberg*).

Per claim 11, *Drake in view of Pastor* and *Lindberg* discloses the method of claim 10, wherein determining a context is implemented using queries written in the Java language or a meta-language (METALANG) or both (See Java taught in paragraph [0035] of *Drake.).*

Per claim 12, *Drake in view of Pastor* and *Lindberg* discloses the method of claim 10, wherein the list of objects forms the context for validation (See Fig. 2 of *Lindberg* and paragraph [0016]).

Per claims 13 and 22, rejection of claims 7, 10, and 19 are fully incorporated. Claims 13 and 22 are rejected under the same rationale as claim 7 due to the similarity in scope in the limitations of the claims. (See respective claims above.)

Per claim 16, the rejection of claims 1 and 7 are incorporated. Further, *Drake in view of Pastor* and *Lindberg* discloses the method of claim 1, wherein each subject in the context is one of an instance of an object, an instance of an object containing an attribute, an instance of an object having an association and an instance of root object of a deployable unit of a collection of objects (See rejection of claims 1 and 7 above).

Per claim 23, rejection of claims 8, 9, and 22 are fully incorporated.

Claim 23 is rejected under the same rationale as claims 8 and 9 due to the similarity in scope in the limitations of the claims. (See respective claims above.)

Per claim 24, rejection of claims 10 and 22 are fully incorporated.

Claim 24 is rejected under the same rationale as claims 10 due to the similarity in scope in the limitations of the claims. (See respective claims above.)

Conclusion

7. **THIS ACTION IS MADE FINAL.** Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the mailing date of this final action.

Contact Information

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Farhan M. Syed whose telephone number is 571-272-7191. The examiner can normally be reached on 8:30AM-5:00 PM.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Neveen Abel-Jalil can be reached on 571-272-4074. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

/F. M. S./
Examiner, Art Unit 2165

/Neveen Abel-Jalil/

Supervisory Patent Examiner, Art Unit 2165